International Journal of Electrical Engineering and Technology (IJEET)
Volume 11, Issue 9, November 2020, pp. 261-273, Article ID: IJEET_11_09_026
Available online at https://iaeme.com/Home/issue/IJEET?Volume=11&Issue=9
ISSN Print: 0976-6545 and ISSN Online: 0976-6553
DOI: 10.34218/IJEET.11.9.2020.026

MAPREDUCE METHODOLOGY FOR ELLIPTICAL CURVE DISCRETE LOGARITHMIC PROBLEMS – SECURING TELECOM NETWORKS

Dr. M. Subhashini[1] and Dr. R. Gopinath[2]

[1]Assistant Professor in Department of Computer Science,
Srimad Andavan Arts and Science College (Autonomous)
Affiliated to Bharathidasan University, Tiruchirappalli, Tamil Nadu, India

[2]D.Litt. (Business Administration)-Researcher, Madurai Kamaraj University,
Madurai, Tamil Nadu, India

ABSTRACT

Elliptic Curve Cryptography (ECC) is capable of constructing public-key crypto-systems. Specifically, the security of the ECC minimizes to testing the ability to handle and solve the DLP (Discrete Logarithmic Problem) in the group of points of an elliptic curve (ECDLP). ECC based on ECDLP is in the list of recommended algorithms for use by NIST (National Institute of Standards and Technology) and NSA (National Security Agency). Given that ECDLP based cryptosystems are in wide-spread utilization, continuous efforts on monitoring the effectiveness of new attacks or improvements to pre-existing attacks on ECDLP over large prime factor is a significant part in that. This paper aims to provide a secure, effective, and flexible method to improve data security in cloud computing. In this chapter, a novel algorithm using MapReduce and Pollard-Rho's approach to solve the ECDLP problems and to enhance the security level.

Key Words: Elliptical Curve Discrete Logarithms Problems, MapReduce approach, Pollard Rho's Approach, Elliptical curve Cryptography, RSA, ElGamal

Cite this Article: M. Subhashini and R. Gopinath, Mapreduce Methodology for Elliptical Curve Discrete Logarithmic Problems – Securing Telecom Networks, *International Journal of Electrical Engineering and Technology (IJEET)*, 11(9), 2020, pp. 261-273.
https://iaeme.com/Home/issue/IJEET?Volume=11&Issue=9

1. INTRODUCTION

The telecommunications network security includes the transmission methods, transport formats, structures and security means which provides integrity confidentiality, authentication and availability for the transmission over the public and private communication media and

network [1]. The information security domain is highly concerned with the data protection for voice and video communication. With the passage of time and advancement in technology the telecommunication network is taking multiple initiates in order to improve their network security [2; 11]. The improvement in security networks boosts the flexibility and it improves the efficiency. It further saves the time and cost and provides efficient business solutions. Despite growing need for telecommunication security, many companies have not adopted efficient infrastructure which results in risk and failure [3; 12]. The security services aim to add security to the system so that various types of security attacks can be encounter.

The security of ECC depends upon a trapdoor function (compute in one direction) so that assumed the discrete logarithm of a random elliptic curve element similar to a publicly known base point infeasible state. This is known as ECDLP (Elliptic Curve Discrete Logarithm Problem) to be computationally infeasible to solve [4]. The difficult part of the DLP is based on the representation of the group of points in ECDLP. The two most popular finite groups are used for discrete logarithm problems are the group of points on an elliptic curve (EC) over a finite field, represented by $E(F_p)$ and the multiplicative group $(Z/Zp)^*$ of integers modulo denoted as a prime factor p. For solving the ECDLP: Pohlig- Hellman algorithm (which reduces the problem to subgroups of prime order), Shanks' baby-step-giant-step method [5], both the rho method and the kangaroo method have parallel versions because of van Oorschot and Wiener, MOV (Menezes-Okamoto-Vanstone) attack using the Weil pairing, Frey-Rueck attack using the Tate pairing, The attacks on anomalous elliptic curves due to Semaev, Satoh-Araki and Smart, Weil descent (for some special finite fields) [6].

Pollards Rho algorithm has a parallel running time to the Baby-Step Giant-Step technique. The rho and kangaroo algorithms needless storage space and can be distributed. Pollards Rho algorithm exploits the parallel technologies such as Cluster computing, GPGPU, and so on to solve the ECDLP [7][8].

But the existing techniques perform and focus on improving a certain part of the process, for example, better random points generation method and faster elliptic curve points operations. There is a lack of system-level study including scalability with real computation infrastructure. Even if the vast amount of computation/storage resources makes use of processing the parallel collision search, and also it has a big challenge to control them effectively and prevent all potential failures/ errors of the processing procedure, to solve ECDLP using MapReduce. MapReduce is the de-facto industry standard for big data applications and verified as a scalable framework for huge data processing.

2. BACKGROUND STUDY

Elliptic curve cryptography is powerful. Calculating the public key from a known private key and base point can be handled easily. On the other hand, extracting the private key from known public key and base point is not an easy task. This is called an Elliptic Curve Discrete Logarithm Problem. The security of elliptic curve cryptography, which is based on the computational hardness of ECDLP, has been extensively studied for decades. The size of the elliptic curve determines the difficulty of the problem. Therefore, the elliptic curve E and the base point P have to be chosen carefully.

The discrete log problem is believed to be hard compared to the exponentiation problem, and the elliptic curve discrete logarithm problem is even harder. This is because of its different algebraic structure, it's complex arithmetic rules to "add" two points on an elliptic curve, and the lack of an index calculus method for the elliptic curve domain. The main reason why ECDLP is "more trusted" than DLP or FP is because DLP and FP can be solved in sub exponential time with index calculus algorithms. Until recently, no similar result had been

obtained for ECDLP: except for exceptional curves and somewhat non-natural families of parameters, the time required to solve was believed to be exponential in the size of the parameters. Schemes and protocols such as the Diffie-Hellman key exchange, Massey-Omura encryption, El- Gamal public-key encryption, and El-Gamal digital signatures and even the Elliptic Curve Digital Signature Algorithm (ECDSA), all use the fact that attempting to solve the ECDLP is a difficult, if not intractable, problem. For example, notice that the security of this system does not rely at all on Alice and Bob finding a secure way to transmit information, but it relies very heavily on Alice and Bob each having private keys that are very, very difficult to retrieve using only their public keys. Eve can only be thwarted if the information that she can intercept is useless. This brings us to the elliptic curve discrete logarithm problem.

Currently, no such algorithm is known so cryptographic systems based on ECC provide a high level of security with relatively small key sizes. However, as we will see, the complicated calculations make ECC somewhat less effective. Note that elliptic curves are not the only mean to create groups where the DLP is hard. The elliptic curve is just the first member of a bigger family of groups, defined as a group structure on the Jacobi surface of specific curves. The next one is called hyperelliptic. But it seems that these constructions do not add as much insecurity as in complexity.

On the other hand, the ECDLP can be efficiently solved on a quantum computer. Of course, it is well known that the integer factorization and the ordinary discrete logarithm problems can also be efficiently solved on a quantum computer [9]. Thus, if large-scale quantum computers are ever built, then all the major families of public-key systems (DL, RSA, ECC) will be insecure.

In cryptography, an attack is a method of solving a problem. Specifically, an attack aims to find a fast method of solving a problem on which an encryption algorithm depends. The known methods of attack on the elliptic curve (EC) discrete log problem that works for all curves are slow, making encryption based on this problem practical. However, several efficient methods for solving the EC discrete log problem for specific types of elliptic curves are known. This means that one should make sure that the curve one chooses for one's encoding does not fall into one of the several classes of curves on which the problem is tractable. When a huge amount of computation/storage resources are used for the existing methods, it is also a big challenge to manage them efficiently and avoid all potential errors/failures of the processing procedure. To fulfill these above gaps, we design an algorithm to solve ECDLP using MapReduce.

2.1. MapReduce Framework

The MapReduce framework contains in variations namely Map step and reduces step. In Map Step, the master node obtains large problem from the input which splits into smaller sub-problems and then distributes it into worker nodes. A worker node may repeat this and leads to a multi-level tree structure. Worker node processed smaller issues and then it provides back to master node. In Reduce step, the Master node obtains the answers to the subproblems and merges into a predefined method while obtain the output/answer to identify the original problem. The MapReduce framework solves fault tolerant since every node in the cluster is expected toward periodically report back with status updates and completed work. When a node remains silent for processing longer than the expected interval, a master node formulates note and reallocates the task to other nodes.

3. POLLARD'S RHO METHOD FOR SOLVING ELLIPTICAL CURVE DISCRETE LOGARITHM PROBLEM

The Pollard concept is to find k which is satisfying Q = [d] P by isolating the group of points on an elliptic curve into threed is joint sets $S_1, S_2,\ and\ S_3$in an equal size [10]. Define the original iteration function on a point R as follows:

$$f(R) = \begin{cases} R \oplus P, if R \in S_1 \\ [2]R, if\ R \in S_2 \\ R \oplus P, if R \in S_3 \end{cases}$$

To begin with a point $R_0 = [a_0]P \oplus [b_0]$Q where $a_0, b_0 \in [1, n-1]$ are arbitrarily selected and make a sequenceR_i by using this function until the collision takes place. The sequence R_i can be expressed in the term of $[a_i]P \oplus [b_i]$Q, where the number $a_i, b_i \in [1, n-1]$ are computed as follows:

$$a_{i+1} = \begin{cases} (a_i + 1) mod\ n, if R_i \in S_1 \\ 2\ a_i\ mod\ n, if\ R_i \in S_2 \\ a_i, if R_i \in S_3 \end{cases}$$

$$b_{i+1} = \begin{cases} b_i, if R_i \in S_1 \\ 2\ b_i\ mod\ n, if\ R_i \in S_2 \\ (b_i + 1) mod\ n, if R_i \in S_3 \end{cases}$$

Given that the number of point's lies on the elliptic curve which form cyclic group is a finite field, this sequence does not become periodic update after applying this function but will start to repeat. Upon detection of a matching, this is$R_i = R_j \oplus [a_i]P \oplus [b_i]Q = \left[a_j\right]P \oplus \left[b_j\right]Q$, if get $\left(b_j - b_i, n\right) = 1$, d can be explained as follows:

$$d = \log_p Q = \left(\frac{a_i - a_j}{b_j - b_i}\right) mod$$

4. PROPOSED MAPREDUCE BASED POLLARD'S RHO METHODOLOGY FOR SOLVING ECDLP

The selection of the iteration function and the distinguishing property affects the performance of the algorithm. Also using specialized libraries to perform multi-precision operation helps in achieving good speedup. This paper discusses the background of elliptic curve DLP in which finite fields are the basic underlying fields on which the cryptosystem is built. The solvability of the elliptic curve cryptosystem reduces to the solvability of the ECDLP problem in that group. Thus, different attacks on the ECDLP have been studied. Dealing with the ECDLP problem based on the large underlying finite field requires large computational resources. Using MapReduce based ECDLP algorithm on the CPU cluster is a better approach to such instances. Solving ECDLP based on the large finite field requires huge resources and special algorithm (or hardware) have to be considered to perform arithmetic's on large integers. Making the use of arithmetic libraries helps in such scenarios. The MapReduce based ECDLP algorithm can be modified to run on a hybrid cluster of open MP. Also, selecting proper iteration function and proper distinguished property as per the parallel platform can result in good speedup.\

4.1. Choosing a Curve

For each of the cryptographic methods depended on the difficulty of the EC discrete log problem, we must begin by choosing an elliptic curve that is not susceptible to the known fast attacks on the discrete log problem. The curve must, therefore, satisfy the following restrictions:

- There exists a large prime p dividing $\#E(F_p)$.
- #E(F_p)≠q (i.e the curve is not anomalous).
- The order of P does not diving $q^k - 1$ for all k such that $1 \leq k \leq C$, where C is a sufficiently large constant so that it is difficult to solve the discrete logarithm problem in $F_p^x C$

4.2. MapReduce for ECDLP

Mapreduce framework performs with data processing works in the variations of Map phase and reduce phase. To fit parallel collision search into MapReduce, in this algorithm also divide the ECDLP solving process into two steps. In MapReduce based approach, both of the two steps are carried out on distributed computation nodes in a parallel manner:

Map Phase: Each Mapper begins from a different point and a distinguished point in a random way to be searched. For mapper i, it runs a random number generator to obtain two random numbers$a_0, b_0 \in \mathbb{Z}_p$and calculates the start point$X_0 = a_0P + b_0Q$. After the mapper, i starts the iteration process to search for a distinguished point. Different approaches have been developed for the iteration process; this algorithm follows Pollard's approach for the iteration process. The group (P) is separated into three subsets$S_1, S_2, and\ S_3$ with parallel size, and mapper i run the iteration process according to the below equation:

$$\begin{cases} X_{j+1} \leftarrow Q + X_j, a_{j+1} \leftarrow a_j, b_{j+1} \leftarrow b_j + 1, if\ X_j \in S_1 \\ X_{j+1} \leftarrow 2X_j, a_{j+1} \leftarrow 2a_j, b_{j+1} \leftarrow 2b_j, \quad if\ X_j \in S_2 \\ X_{j+1} \leftarrow P + X_j, a_{j+1} \leftarrow a_j + 1, a_{j+1} \leftarrow b_j, if\ X_j \in S_3 \end{cases}$$

The membership judgment has to simplify and use for Hamming weights of points to divide (P) into$S_1, S_2, and\ S_3$:

$$S_k = \{T \in (p) | (hw(T) mod 3) = k - 1\}, k = 1,2,3$$

The sum of the Hamming weights of *T*'s measures two coordinate values using *hw*(*T*), and if *T* = *O*, *hw*(*T*) is set to be0. At the end of j[th] iteration (j=1,2,…), mapper i checks whether X_{j+1} is a distinguished point.

As a collision search job will be detected with distinguished points, the more distinguished points are collected so that collision search job will be detected. This algorithm phase utilizes the Hamming weight of an elliptic curve point to determine whether it is a distinguished point or not. If the Hamming weight is less than *z*, classify it as a distinguished point. The parameter *z* can be tweaked under available sources. If $X_{j+1}$is a distinguished point, *mapper i* outputs akey-value pair in the following form

$$\left[h(X_{j+1}), (X_{j+1}, a_{j+1}, b_{j+1})\right]$$

This procedure generates and continuously repeats until a new random start point. Here *hw*(*T*) is a pre-defined function for *key* generation, and $(X_{j+1}, a_{j+1}, b_{j+1})$ is the value. If X_{j+1} is not a distinguished point, mapper i continue to the next iteration process accordance to the given equation.

Reduce Phase: Reducers were taken responsible for storing generated distinguished points and detection of collisions among these points. To compare with mappers function and their computation burden is much lesser. After receiving a key-value pair [H(X), (X, c, d)], reducer i saves the triple (X,c, d) to the storage system and checks whether there is a pre-existing triple(X', c', d') such that $X = X'$. If such a collision is detected, reduce i can evaluate the value of l as

$$l = \frac{d - d'}{c' - c} \; mod \; p$$

and immediately terminates the whole collision search job.

4.3. Diffie Hellman Key Exchange

The following series of steps describes the Diffie Hellman Key Exchange:

- Alice and Bob publicly agree on $E(\mathbb{F}_p)$, chosen so that the discrete log problem is hard, as described above. They also agree on point $P \in E(\mathbb{F}_p)$ of high (usually prime) order.
- Alice chooses a secret $a \in \mathbb{Z}$, computes aP, and sends it to Bob.
- Bob chooses a secret $b \in \mathbb{Z}$, computes bP, and sends it to Alice.
- Alice computes a(bP) = abP
- Bob computes b(aP) = abP
- Alice and Bob now have the same point abP. They use a publicly agreed on method, such as taking the last 256 bits of the y-coordinate of the point, to extract a key.

4.3. Message Encryption

Message m is considered as a point P_m with coordinates (x,y) in the elliptic curve P_m. The point P_m is encrypted as a ciphertext C_m and subsequently decrypted.

To encrypt and send a messageP_m to Bob, Alice chooses a random positive integer k and produces the ciphertextC_m consisting of the pair of points:

$$C_m = \{kG, P_m + KP_b\}$$

KP_b is a public key of bob.

4.4. Message Decryption

Bob decrypts the ciphertext by multiplying the first point in the pair by Bob's secret key and subtracts the result from the second point:

$$P_m + kP_B - n_B(KG) = \; P_m + k(n_B G) - n_B(kG) = \; P_m$$

Alice has marked the message P_m by adding kP_B to it.

5. EXPERIMENTAL RESULT AND DISCUSSION

The performance of the proposed MR-PR based ECC method is analysed with Twitter dataset. The performance metrics like Key Calculate Time (in milliseconds), Key Updation time (in milliseconds), Throughput (mbps), Encryption Time (in seconds), Decryption Time (in seconds), and Total time taken (in seconds). In this section, the performance of the proposed techniques like MapReduce-Pollard Rho's (MR-PR) method with the existing encryption techniques like ECC, RSA, and ElGamal.

Table 1 Key Calculation Time (in Milliseconds) by Proposed MR-PR Method, ECC, RSA and ElGamal Encryption techniques

Dataset Size (MB	Key Calculation Time (in Milliseconds)			
	Proposed MR-PR Method	**ECC**	**RSA**	**ElGamal**
20	567	646	821	832
40	604	692	878	881
60	653	743	935	942
80	721	803	992	990
100	786	868	1053	1052
120	814	921	1121	1128
140	863	986	1198	1196
160	901	1072	1263	1262
180	924	1142	1314	1322
200	998	1210	1382	1381

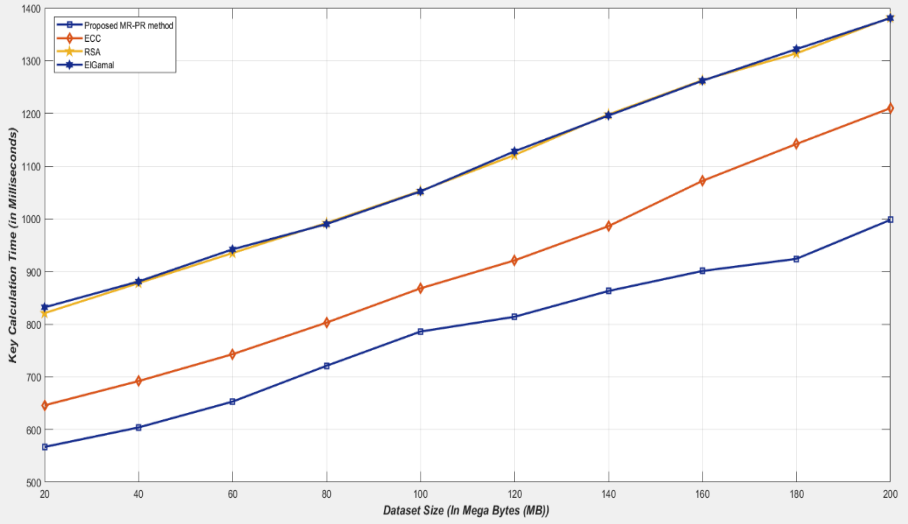

Figure 1 Graphical representation of the Key Calculation time (in Milliseconds) by the proposed MR-PR method, ECC, RSA and ElGamal techniques

Table 1 gives the Key calculation time (in milliseconds) by the proposed MR-PR method, ECC, RSA and ElGamal techniques with varying dataset size (in MB). Figure 1 gives the graphical representation of the key calculation time (in milliseconds) by the proposed MR-PR method, ECC, RSA and ElGamal techniques. From the table 1 and figure 1, it is clear that the proposed MR-PR method generates the key in least time when it is compared with existing techniques. The proposed MR-PR method also gives the key in least time when the dataset size is increasing. The encryption techniques like RSA and ElGamal have the consistent key generation time than the proposed MR-PR method and ECC.

Table 2 Key Updation Time (in Milliseconds) by Proposed MR-PR Method, ECC, RSA and ElGamal Encryption techniques

Dataset Size (MB	Key Updation Time (in Milliseconds)			
	Proposed MR-PR Method	ECC	RSA	ElGamal
20	239	342	392	421
40	248	385	463	498
60	261	438	518	532
80	275	503	592	603
100	283	592	684	685
120	297	665	735	745
140	310	732	821	843
160	321	848	906	918
180	342	931	998	1015
200	356	928	1024	1035

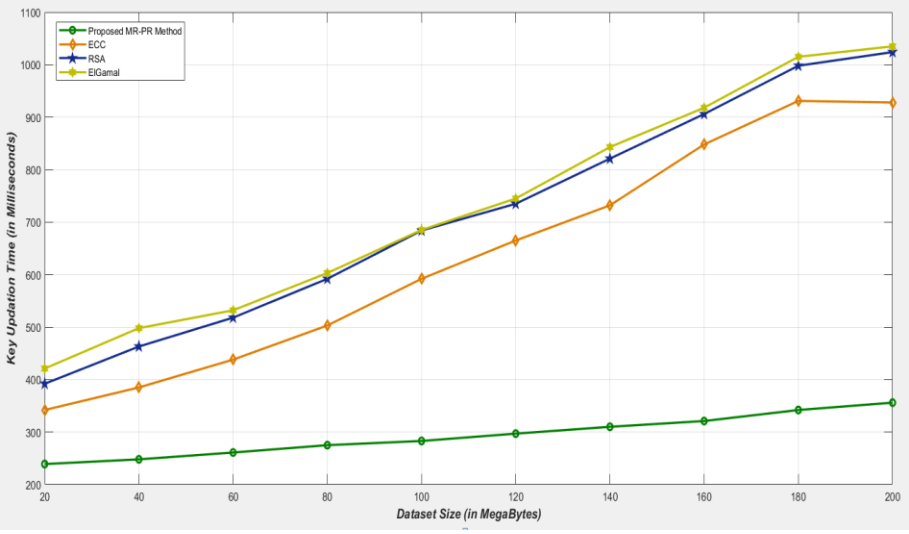

Figure 2: Graphical representation of the key updation time (in milliseconds) by the proposed MR-PR method, ECC, RSA and ElGamal techniques

Table 2 gives the Key updation time (in milliseconds) by the proposed MR-PR method, ECC, RSA and ElGamal techniques with varying dataset size (in MB). Figure 2 gives the graphical representation of the key updation time (in milliseconds) by the proposed MR-PR method, ECC, RSA and ElGamal techniques. From the table 2 and figure 2 it is clear that the proposed MR-PR method updates the key in least time when it is compared with existing techniques. The proposed MR-PR method also updates the key in least time when the dataset size is increasing. The encryption techniques like RSA and ElGamal have the consistent key updation time than the proposed MR-PR method and ECC.

Table 3 Encryption Time (in Seconds) by Proposed MR-PR Method, ECC, RSA and ElGamal Encryption techniques

Dataset Size (MB	Encryption Time (in Seconds)			
	Proposed MR-PR Method	ECC	RSA	ElGamal
20	28	43	68	69
40	34	68	83	88
60	49	81	96	98
80	63	102	118	128
100	82	123	138	145
120	98	142	167	175
140	109	168	181	192
160	122	185	203	214
180	141	204	234	245
200	156	235	263	275

Table 3 gives the Encryption time (in seconds) by the proposed MR-PR method, ECC, RSA and ElGamal techniques with varying dataset size (in MB). Figure 3 gives the graphical representation of the Encryption time (in seconds) by the proposed MR-PR method, ECC, RSA and ElGamal techniques. From the table 3 and figure 3 it is clear that the proposed MR-PR method done the encryption in least time when it is compared with existing techniques. The proposed MR-PR method also encrypts the dataset in least time when the dataset size is increasing. The encryption techniques like RSA and ElGamal have the consistent encryption time than the proposed MR-PR method and ECC.

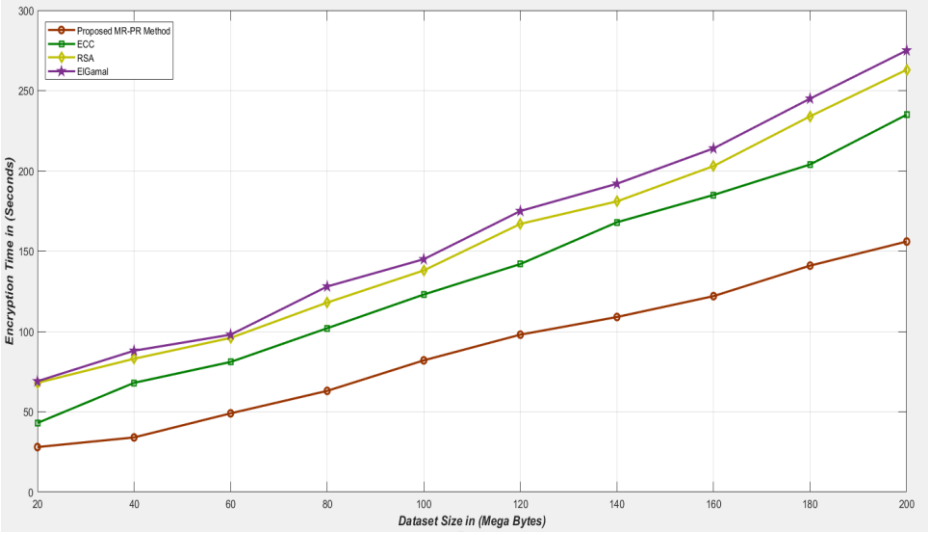

Figure 3 Graphical representation of the Encryption time (in seconds) by the proposed MR-PR Method, ECC, RSA and ElGamal techniques

Table 4 gives the Decryption time (in seconds) by the proposed MR-PR method, ECC, RSA and ElGamal techniques with varying dataset size (in MB). Figure 4 gives the graphical representation of the Decryption time (in seconds) by the proposed MR-PR method, ECC, RSA and ElGamal techniques. From the table 4 and figure 4, it is clear that the proposed MR-PR method done the decryption in least time when it is compared with existing techniques. The proposed MR-PR method also decrypts the dataset in least time when the dataset size is

increasing. The encryption techniques like RSA and ElGamal have the consistent decryption time than the proposed MR-PR method and ECC.

Table 4 Decryption Time (in Seconds) by Proposed MR-PR Method, ECC, RSA and ElGamal Encryption techniques

Dataset Size (MB	Decryption Time (in Seconds)			
	Proposed MR-PR Method	ECC	RSA	ElGamal
20	21	32	44	53
40	29	48	53	60
60	42	67	72	75
80	54	75	83	90
100	69	92	95	102
120	82	118	123	128
140	96	134	145	150
160	112	156	165	173
180	128	178	187	192
200	132	192	198	204

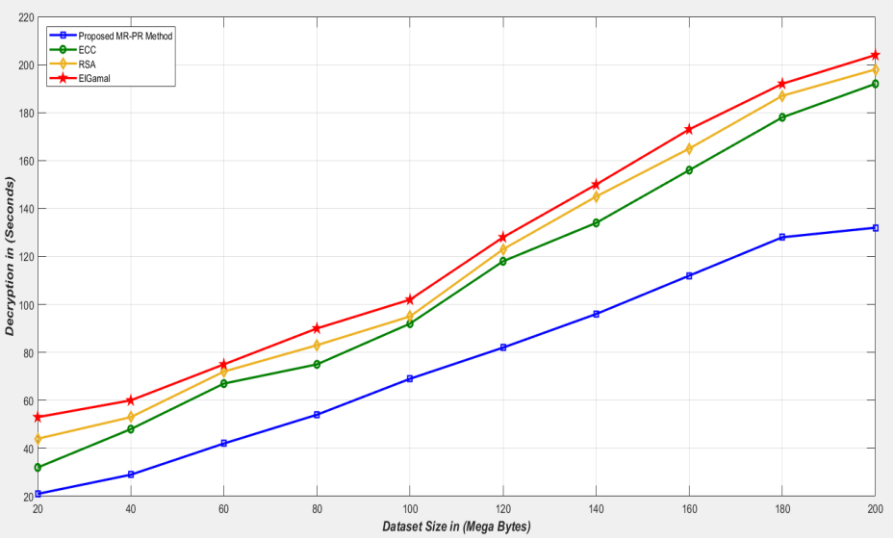

Figure 4 Graphical representation of the decryption time (in seconds) by the Proposed MR-PR Method, ECC, RSA and ElGamal techniques

Table5 gives the total encryption and decryption time (in seconds) by the proposed MR-PR method, ECC, RSA and ElGamal techniques with varying dataset size (in MB). Figure 5 gives the graphical representation of the total encryption and decryption time (in seconds) by the proposed MR-PR method, ECC, RSA and ElGamal techniques. From the table 5 and figure 5, it is clear that the proposed MR-PR method done the total encryption and decryption time in least time when it is compared with existing techniques. The proposed MR-PR method also encrypts and decrypts the dataset in least time when the dataset size is increasing. The encryption techniques like RSA and ElGamal have the consistent total encryption and decryption time than the proposed MR-PR method and ECC.

Table 5 Total Encryption/Decryption Time taken (in Seconds) by Proposed MR-PR Method, ECC, RSA and ElGamal Encryption techniques

Dataset Size (MB	Total Encryption/Decryption Time (in Seconds)			
	Proposed MR-PR Method	ECC	RSA	ElGamal
20	49	75	112	122
40	63	116	136	148
60	91	148	168	173
80	117	177	201	218
100	157	215	233	247
120	180	260	290	303
140	205	302	326	342
160	234	314	368	387
180	264	382	421	437
200	294	427	461	479

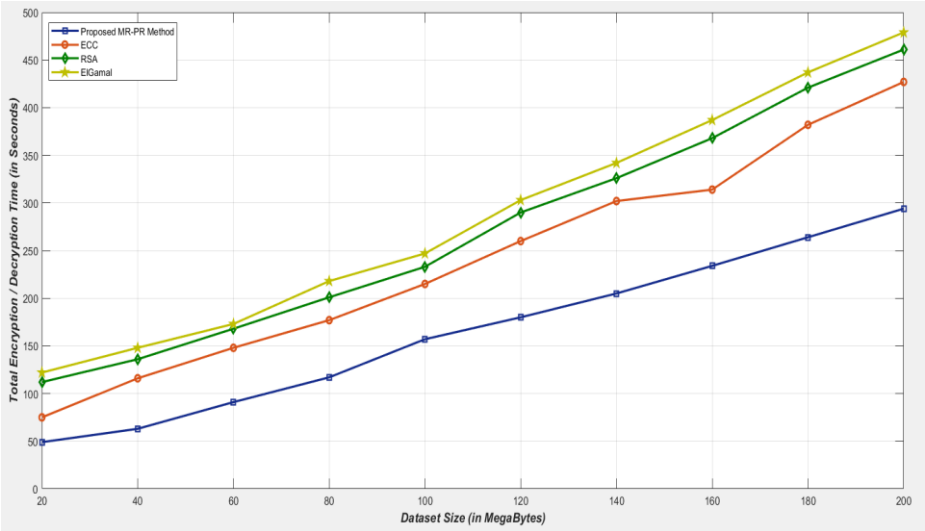

Figure 5 Graphical representation of the total encryption and decryption time taken (in seconds) by the proposed MR-PR method, ECC, RSA and ElGamal techniques

Table 6 gives the throughput (in mbps) by the proposed MR-PR method, ECC, RSA and ElGamal techniques with varying dataset size (in MB). Figure 6 gives the graphical representation of the throughput (in mbps) by the proposed MR-PR method, ECC, RSA and ElGamal techniques. From the table 6 and figure 6, it is clear that the proposed MR-PR method gives the high throughout rate when it is compared with existing techniques. The proposed MR-PR method also gives the throughput for encryption and decryption of the dataset is high when the dataset size is increasing. The encryption techniques like RSA and ElGamal have the consistent throughput than the proposed MR-PR method and ECC.

Table 6 Throughput (in mbps) by Proposed MR-PR Method, ECC, RSA and ElGamal Encryption techniques

Dataset Size (MB	Throughput (in mbps)			
	Proposed MR-PR Method	ECC	RSA	ElGamal
20	1436	1342	1128	1113
40	1487	1381	1186	1172
60	1521	1415	1245	1230
80	1545	1464	1288	1273
100	1598	1532	1326	1314
120	1628	1583	1378	1368
140	1675	1641	1432	1426
160	1708	1692	1483	1467
180	1734	1714	1535	1518
200	1789	1728	1590	1575

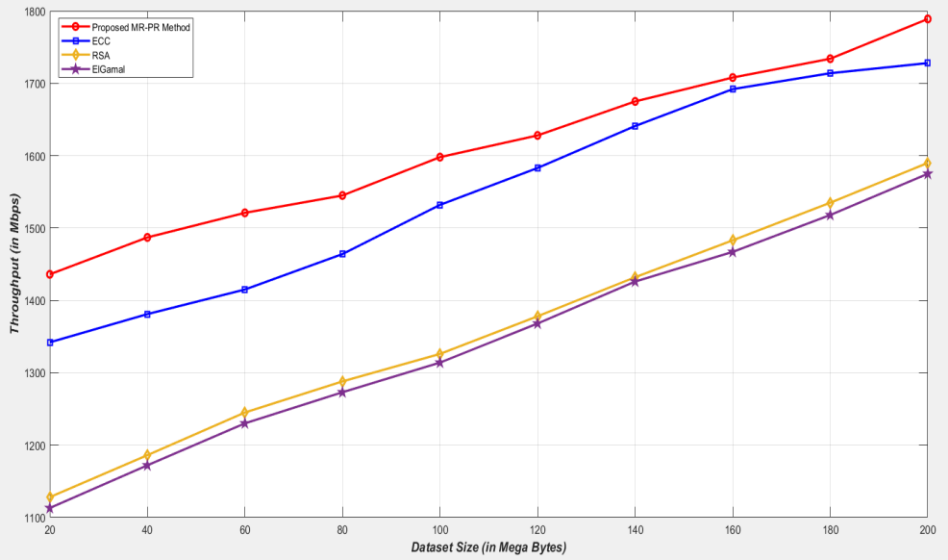

Figure 6 Graphical representation of the Throughput (in mbps) by the proposed MR-PR Method, ECC, RSA and ElGamal techniques

6. CONCLUSION

In this paper, solving the Elliptical Curve Discrete Logarithm problem in the ECC, a MapReduce-Pollard Rho's based ECC method is proposed. The result obtained by the proposed method is depicted in this chapter. The metrics like Encryption time, Decryption time, throughput, key calculation, key updation and total time for encryption and decryption is considered. In this paper, the result obtained by the proposed MR-PR based ECC method is projected with the Twitter dataset. From the results and discussion, it is clear that the proposed MR-PR based ECC method performed with Less encryption, decryption, key generation, key updation, and total encryption/decryption time and improved throughput than the existing security algorithms like RSA, ECC and Elgamal.

REFERENCES

[1] Gopinath, R. "A Study on Recruitment and Selection in Bsnl with Special Reference to Job Satisfaction in Three Different Ssas Using Sem Modeling." Management 5.7 (2016).

[2] Gopinath, R. "How the Compensation Management and Welfare Measure Influence Job Satisfaction? A Study with Special Reference in BSNL to Three Different SSAs Using Modeling." Management 5.8 (2016).

[3] Gopinath, R. (2016). A Study on Training and Development in BSNL with special reference to Job Satisfaction in three different SSAs using Modeling. Global Journal for Research Analysis, 5(6), pp. 367-370

[4] M. J. Wiener and R. J. Zuccherato, "Faster Attacks on Elliptic Curve Cryptosystem", In selected areas in Cryptography- SAC, volume 1556 of LNCS, pages 190-200, Springer, 1999.

[5] Menezes A. J., Okamoto T., Vanstone S. A, "Reducing Elliptic Curve Logarithms to Finite field", IEEE Trans. Info. Theory, 39: 1639-1646, 1993.

[6] P. C. van Oorschot and M. J. Wiener, "Parallel collision search with cryptanalytic applications," Journal of Cryptology, vol. 12, no. 1, pp. 1–28, 1999.

[7] E. Wenger and P. Wolfger, "Solving the Discrete Logarithm of a 113- bit Koblitz Curve with an FPGA cluster", SAC, volume 8781 of LNCS, pages 363-379, Springer, 2014.

[8] P. C. van Oorshant and M. J. Wiener, "Parallel Collision Search with Cryptanalytic Applications", Journal of Cryptology, Volume 12 No. 1, pages 1-28, 1999.

[9] Alfred J. Menezes, Paul C. van Oorschot, and Scott A. Vanstone., "Handbook of applied cryptography", CRC Press Series on Discrete Mathematics and its Applications, CRC Press, Boca Raton, FL, 1997., With a foreword by Ronald L. Rivest.

[10] R. Gallant, R. Lambert, and S. Vanstone, "Improving the parallelized Pollard lambda search on anomalous binary curves", Mathematics of Computation of the American Mathematical Society, volume 69 No. 232, pages 19-46, 2002.

[11] Gopinath, R. "Impact of HRD to Job Satisfaction with special reference to BSNL Employees In three different SSAs using SEM Model." International Journal of Management (IJM) 7.5 (2016): 1-9.

[12] Gopinath, R. "Is the Employee Health and Safety related to Job Satisfaction? An inquiry into BSNL Employees with special reference in three different SSAs using Modeling." IOSR Journal of Business and Management 18.7 (2016): 135-139.